Amendments to the Claims

The listing of claims below will replace all prior versions, and listings, of claims in the application. Please amend the claims as follows:

Listing of Claims:

1. (Currently Amended) A system for printing data on a printer comprising:

a) ~~one or more~~a plurality of clients that include a plurality of application[[s]] programs having a print capability;

b) a server that implements a server print spooler for coordinating print requests~~the printing of data~~ originating from the plurality of application programs and communicated to said server print spooler by the ~~one or more~~ plurality of clients, the server print spooler comprising a thread manager that maintains a first thread pool for accepting the print requests and a second thread pool for processing the print requests, the second thread pool being implemented by a completion port that limits the number of threads in the second thread pool used for processing the print requests, wherein the plurality of clients use asynchronous remote procedure calls to communicate the print requests to the server print spooler and when the server print spooler receives an asynchronous remote procedure call with a print request from a client, the server transmits a first response to the client in order to prevent blocking an application program thread originating the print request from performing other processing, and the server transmits a second response to the client after the print request is processed; and

c) ~~one or more~~a plurality of printers coupled to the server for printing under direction of the server print spooler~~wherein a clients includes a client print spooler running that communicates with at least one of said one or more printers by asynchronous remote procedure calls to the server print spooler~~.

2. (Canceled) ~~The system of claim 1 wherein said server spooler includes a thread manager for maintaining a thread pool for servicing pending client print requests by communicating data to the plurality of printers.~~

3. (Canceled) ~~The system of claim 1 wherein the server print spooler implements a completion port wherein incoming print requests from the clients are added to the completion port~~.

4. (Currently Amended) The system of claim 1 wherein the server print spooler maintains a list of print requests awaiting servicing and wherein the list of print requests are added to [[a]]the second thread pool that shares processor time of the server.

5. (Currently Amended) The system of claim 4 wherein the first thread pool and the second thread pool are each ~~multiple thread pools are created to service the print request and each thread pool is~~ serviced by a single processor thread.

6. (Original) The system of claim 1 wherein the client print spooler implements certain procedures asynchronously and some of said procedures are implemented in a synchronous manner.

7. (Original) The system of claim 1 wherein the clients send data to the server print spooler in multiple asynchronous requests until an entire print job is completed.

8. (Currently Amended) The system of claim 1 additionally comprising a scheduler for sending print requests to [[a]] the second thread pool for processing.

9. (Currently Amended) The system of claim 8 wherein the scheduler chooses print requests from [[a]]the first ~~print processing~~ thread pool and adds them to [[a]] the second ~~print processing~~ thread pool.

10. (Currently Amended) The system of claim 8 wherein the scheduler sends raw data to the second thread pool in an overlapped manner.

11. (Currently Amended) A method of printing data originating from a plurality of clients on a plurality of printers comprising:

a) receiving an asynchronous remote procedure call with a print request from one of the plurality of clients, the asynchronous remote procedure call originating from an application thread on the one of the plurality of clients and being communicated to a server using a print spooler interface on a print server;

b) in response to the receiving, returning a first response to the client before the print request is processed in order to prevent blocking the application thread from performing other processing;

c) scheduling the print request using a scheduler;

d) processing the print request, wherein the scheduler allocates print server run time to the print request for processing the print request; and

e) returning a second response to the client after the print request is processed.

~~a) providing a print spooler interface for an application to communicate with a client which in turn communicates with a print server; said print spooler interface enabling the one or more applications to call a service routine on the print server by means of an asynchronous remote procedure call originating with the one or more applications; and~~

~~b) implementing a scheduler on the print server computer that responds to the request from the client for print services by allocating print server run time amongst print requests sent to the server by one or more computer applications.~~

12. (Currently Amended) The method of claim 11, wherein the scheduler uses ~~wherein the scheduler implements~~ one or more thread pools to schedule and process ~~service~~ the print requests~~ from the multiple computer applications~~.

13. (Original) The method of claim 12 wherein the one or more thread pools are implemented by a completion port wherein incoming print requests from the client computers are added to the completion port.

14. (Currently Amended) The method of claim 11 wherein the print server maintains a list of print requests awaiting servicing and wherein the list of print requests are added to a thread pool that shares processor time of the [[a]] print server computer.

15. (Original) The method of claim 14 wherein multiple thread pools are created to service the print requests and each thread pool is serviced by a single processor thread.

16. (Canceled) ~~The method of claim 11 wherein the client print spooler implements certain procedures asynchronously and some of said procedures are implemented in a synchronous manner~~.

17. (Currently Amended) The method of claim 11 wherein the ~~client~~ print request is ~~a print job~~ made up of multiple print requests which are sequentially communicated to the print server computer in multiple asynchronous remote procedure calls.

18. (Currently Amended) The method of claim 11 wherein the[[a]] scheduler sends print requests to a thread pool for processing.

19. (Original) The method of claim 18 wherein the scheduler chooses print requests from a first print processing thread pool and adds the requests to a second print processing thread pool.

20. (Currently Amended) In a client/server computing system, a method for implementing a server print spooler comprising:

a) receiving ~~asynchronous client~~ asynchronous remote procedure calls with print requests by means of a communications channel that conveys print requests to a server print spooler;

b) for each print request received, returning a first response to an application thread originating the print request before scheduling print output for the print request in order to prevent blocking the application thread from performing other processing;

c[[b]]) placing the ~~client~~ print requests into a queue of such print requests; ~~and~~

d[[c]]) scheduling print output from a subset of said print requests at a selected print location by placing the subset of said print requests in a thread pool serviced by a processor thread which switches between servicing the print requests in the thread pool.

21. (Currently Amended) A computer readable medium for implementing a print spooler for printing data originating from a plurality of clients on a plurality of printers comprising instructions for:

a) providing a print spooler interface for a computer application on one of the plurality of clients to communicate with a server; said print spooler interface enabling [[the]] a computer~~one or more~~ application[[s]] thread to call a service routine on a print server to send a print request by means of an asynchronous remote procedure call ~~originating with said application~~; and

b) implementing a scheduler on the print server that responds to the print request from the computer application~~client~~ for print services to allocate server run time amongst a plurality of print requests sent to the server by multiple computer applications, wherein the scheduler in response to receiving the print request, returns a first response to the computer application thread before the print request is processed in order to prevent blocking the computer application thread from performing other processing and returns a second response to the computer application indicating the print request has been processed.

22. (Currently Amended) The computer readable medium of claim 21 wherein the scheduler comprises a thread pool [[is]] implemented by a completion port wherein incoming print requests from applications on of the plurality of clients ~~a client~~ are added to the completion port.

23. (Original) The computer readable medium of claim 21 wherein the print server maintains a list of print requests awaiting servicing and wherein the list of print requests are added to a thread pool that shares processor time of the server.

24. (Original) The computer readable medium of claim 23 wherein multiple thread pools are created to service the print requests and each thread pool is serviced by a single processor thread.

25. (Original) The computer readable medium of claim 21 wherein the print spooler interface implements certain procedures asynchronously and some of said procedures are implemented in a synchronous manner.

26. (Currently Amended) The computer readable medium of claim 21 wherein the scheduler is capable of processing a print job made up of multiple print requests which are sequentially communicated to the print server in multiple asynchronous remote procedure calls.

27. (Currently Amended) The computer readable medium of claim 21 wherein [[a]]the scheduler sends print requests to a thread pool for processing.

28. (Original) The computer readable medium of claim 27 wherein the scheduler chooses print requests from a first print processing thread pool and adds the requests to a second print processing thread pool.